



Table of Contents

Load Searching Web Service URL.....	2
Production.....	2
Test.....	2
Load Searching Web Service Requests	2
GetLoadSearchResults: Searches for loads based on search criteria.	2
GetLoadSearchDetailResult: Get the details for a specified load.	4
GetMultipleLoadDetailResults: Get the details for loads based on search criteria.....	5
Appendix 1 – SOAP Object Descriptions	6
class LoadSearchRequest	6
class LoadSearchCriteria	6
class LoadSearchReturn	7
class LoadSearchItem	7
class LoadDetailRequest	7
class LoadDetailReturn.....	8
class MultipleLoadDetailReturn	8
class LoadDetailResult.....	8
class MultipleLoadDetailResult	9
class Error.....	10
class EquipmentTypes.....	10
enum LoadType.....	10
enum TrailerOptionType.....	10
enum SortColumns	11
Appendix 2 – Supported Equipment Types	11
Appendix 3 – Supported Countries.....	13
Appendix 4 - Supported States and Provinces.....	13
United States.....	13
Canada	15
Mexico.....	15
Appendix 5 – Bond Information.....	16

Appendix 6 – XML Examples	17
GetLoadSearchResults XML	17
GetLoadSearchDetailResult XML	19
GetMultipleLoadDetailResults XML	21

Load Searching Web Service

Load Searching Web Service URL

Production

<http://webservices.truckstop.com/V13/Searching/LoadSearch.svc>

Test

<http://testws.truckstop.com:8080/V13/Searching/LoadSearch.svc>

Production URL can be either https(Secure) or http.

Load Searching Web Service Requests

GetLoadSearchResults: Searches for loads based on search criteria.

Input

LoadSearchRequest

Requirements

Authentication requires an account with a valid user name, password, and integration ID that has Load Searching Web Service enabled.

Load Searching requires the following:

Origin – specified as a {State}, {City and State}, {City, State, and Country}, or {latitude and longitude} in integer format. For example, Boise, ID has a latitude of 43.60 and longitude of -116.21. To get these in integer formats, simply multiply by 100. So Boise would be represented by a latitude of 4360 and longitude of -11621. Range is optional and if included needs to be within 25 and 300. If not specified, it will default to 25. For an anywhere search, leave city and state blank and pass country. For multi-state searches, enter a comma-delimited list of state abbreviations in the State field. (See **Appendix 3 – Supported Countries** for valid OriginCountry and DestinationCountry values and see **Appendix 4 – Supported States and Provinces** for supported states and provinces)

Destination –specified as a {State}, {City and a State}, {City, State and Country}, or {latitude and longitude} in integer format.

EquipmentType - a comma-delimited list specifying between 1 to 3 different equipment-types. Specifying “ANY” will search for all types. See [Appendix 2 – Supported Equipment Types](#) for the recognized equipment-type tags.

Load Searching Notes

To search for loads based on Age, specify the age in hours in HoursOld field. If filtering based on Age is not required, specify a 0 in the HoursOld field.

Paging functionality is available to specify the number of results per page and to select a page from the results. PageNumber specifies which page to select. Entering 0 will ignore the PageSize limit and return all results. PageSize specifies how many rows to return and must be a number not greater than 200.

Sort functionality is available to specify a column to sort on and the sort order. The sort column is a specified list of column names. See SortColumns enum. The sort order is a Boolean value that indicates Ascending order when false, and Descending order when true.

Output

LoadSearchReturn

Sample Code - .NET 3.5 using C#

Search for a load with an origin within 200 miles of Boise, Idaho, any destination, equipment type Van, and all Load Types, Sorted by Origin State Descending

```
LoadSearchClient client = new LoadSearchClient();
LoadSearchRequest request = new LoadSearchRequest();
request.IntegrationId = 12345;
request.UserName = "xxxxx";
request.Password = "xxxxx";

request.Criteria = new LoadSearchCriteria
{
    OriginCity = "Boise",
    OriginState = "ID",
    OriginRange = 200,
    EquipmentType = "V,R",
    LoadType = LoadType.All,
    SortBy = SortColumn.OriginCity,
    SortDescending = true
};

LoadSearchReturn result = client.GetLoadSearchResults(request);

foreach (Error error in result.Errors)
{
    Console.WriteLine(error.ErrorMessage);
}

foreach (LoadSearchItem item in result.SearchResults)
{
    Console.WriteLine(item.OriginCity + "," + item.OriginState));
    Console.WriteLine(item.DestinationCity + "," + item.DestinationState));
}
```

Sample Code - PHP 5

Search for a load with an origin within 100 miles of Boise, Idaho to Seattle, WA, equipment type Van, all Load Types with a Pickup Date of Jan 1st, 2011

```
$wsdl_url = 'http://testws.truckstop.com:8080/V13/Searching/LoadSearch.svc?wsdl';
$client = new SOAPClient($wsdl_url);
$params = array(
    'searchRequest' => array(
        'UserName' => $username, 'Password' => $password, 'IntegrationId' => $intID,
        'Criteria' => array(
            'OriginCity' => 'Boise', 'OriginState' => 'ID', 'OriginCountry' => 'USA',
            'OriginRange' => 100, 'OriginLatitude' => '', 'OriginLongitude' => '',
            'DestinationCity' => 'Seattle', 'DestinationState' => 'WA', 'DestinationCountry' =>
            'USA', 'DestinationRange' => 100, 'EquipmentType' => 'F,V', 'LoadType'
            => 'All', 'PickupDate' => '2011-01-01T00:00:00-06:00', 'EquipmentOptions' => ''
        )
    )
);
$return = $client->GetLoadSearchResults($params);

echo "\nErrors:\n";
print_r($return->GetLoadSearchResultsResult->Errors);

$searchResults = $return->GetLoadSearchResultsResult->SearchResults->LoadSearchItem;

$resultSize = sizeof($searchResults);
echo "\nTotal Number of Results: $resultSize \n";
print_r($searchResults);
```

GetLoadSearchDetailResult: Get the details for a specified load.

Input

LoadDetailRequest

Requirements

Authentication requires an account with a valid user name, password, and integration ID that has Load Searching Web Service enabled.

LoadID – the ID of the load you want to retrieve detail for.

Output

LoadDetailReturn

Sample Code - .NET 3.5 using C#

Retrieve load details for Load ID 12346789

```
LoadSearchClient client = new LoadSearchClient();
LoadDetailRequest request = new LoadDetailRequest();
request.IntegrationId = 12345;
request.UserName = "xxxxx";
request.Password = "xxxxx";
request.LoadId = 123456789;

LoadDetailReturn result = client.GetLoadSearchDetailResult(request);

foreach (Error error in result.Errors)
{
    Console.WriteLine(error.ErrorMessage);
}
```

```
Console.WriteLine(result.LoadDetail.TruckCompanyName);  
Console.WriteLine(result.LoadDetail.TruckCompanyPhone);
```

GetMultipleLoadDetailResults: Get the details for loads based on search criteria.

Input

LoadSearchRequest

Requirements

Authentication requires an account with a valid user name, password, and integration ID that has Load Posting Web Service enabled.

Load Searching requires the following:

Origin – specified as a {State}, {City and State}, {City, State, and Country}, or {latitude and longitude} in integer format. For example, Boise, ID has a latitude of 43.60 and longitude of -116.21. To get these in integer formats, simply multiply by 100. So Boise would be represented by a latitude of 4360 and longitude of -11621. Range is optional and if included needs to be within 25 and 300. If not specified, it will default to 25. For an anywhere search, leave city and state blank and pass country. For multi-state searches, enter a comma-delimited list of state abbreviations in the State field.(See [Appendix 3 – Supported Countries](#) for valid OriginCountry and DestinationCountry values and see [Appendix 4 – Supported States and Provinces](#) for supported states and provinces)

Destination – specified as a {State}, {City and a State }, {City, State and Country}, or {latitude and longitude} in integer format.

Output

MultipleLoadDetailReturn

Load Searching Notes

To search for loads based on Age, specify the age in hours in HoursOld field. If filtering based on Age is not required, specify a 0 in the HoursOld field.

Paging functionality is available to specify the number of results per page and to select a page from the results. PageNumber specifies which page to select. Entering 0 will ignore the PageSize limit and return all results. PageSize specifies how many rows to return and must be a number not greater than 200.

Sort functionality is available to specify a column to sort on and the sort order. The sort column is a specified list of column names found in Appendix 7. The sort order is a Boolean value that indicates Ascending order when false, and Descending order when true.

Sample Code - .NET 3.5 using C#

Retrieve load details for Load ID 12346789

```
LoadSearchClient client = new LoadSearchClient();  
LoadSearchRequest request = new LoadSearchRequest();  
request.IntegrationId = 12345;  
request.UserName = "xxxxx";  
request.Password = "xxxxx";  
  
request.Criteria = new LoadSearchCriteria  
{  
    OriginCity = "Boise",  
    OriginState = "ID",
```

```

        OriginRange = 200,
        EquipmentType = "V",
        LoadType = LoadType.All
    };

    MultipleLoadDetailReturn results = client.GetMultipleLoadDetailResults(request);

    foreach (Error error in results.Errors)
    {
        Console.WriteLine(error.ErrorMessage);
    }

    foreach (var load in results.DetailResults)
    {
        Console.WriteLine(load.TruckCompanyName);
        Console.WriteLine(load.TruckCompanyPhone);
    }

```

Appendix 1 – SOAP Object Descriptions

(* required)

(VA – Value Add)

class LoadSearchRequest

string	UserName	*
string	Password	*
int	IntegrationId	*
LoadSearchCriteria	Criteria	*

class LoadSearchCriteria

string	OriginCity	
string	OriginState	
string	OriginCountry	
int	OriginRange	(within 25 and 300)
int	OriginLatitude	(Optional)
int	OriginLongitude	(Optional)
string	DestinationCity	
string	DestinationState	
string	DestinationCountry	
int	DestinationRange	(within 25 and 300)
string	EquipmentType	A comma-delimited list specifying between 1 to 3 different equipment-types. See Appendix 2 – Supported Equipment Types for the recognized tags.
LoadType	LoadType	
DateTime[]	PickupDate	
Int	HoursOld	
TrailerOptionType[]	EquipmentOptions	
int	PageNumber	(Must be greater than zero)

int	PageSize	(Maximum of 200)
SortColumn	SortBy	
Bool	SortDescending	

class LoadSearchReturn

Error[]	Errors	
LoadSearchItem[]	SearchResults	(Sorted by ENTERED descending)

class LoadSearchItem

int	ID	
string	PointOfContactPhone	
string	Equipment	
TrailerOptionType[]	EquipmentOptions	
string	Age	
LoadType	LoadType	
string	PickUpDate	
string	OriginCity	
string	OriginState	
string	OriginCountry	
int	OriginDistance	
string	DestinationCity	
string	DestinationState	
string	DestinationCountry	
int	DestinationDistance	
string	Payment	
string	Length	
string	Weight	
string	CompanyName	
string	Miles	
bool	BondEnabled	– see Appendix 5 – Bond Information for details
int	Bond	– see Appendix 5 – Bond Information for details
int	BondTypeID	– see Appendix 5 – Bond Information for details
bool	IsFriend	– VA
decimal	PricePerGall	– VA
string	FuelCost	– VA
string	ExperienceFactor	– VA
string	Days2Pay	– VA

class LoadSearchResults

Collection<LoadSearchItem>

class LoadDetailRequest

string	UserName	*
string	Password	*
int	IntegrationId	*
int	LoadId	*

class LoadDetailReturn

LoadDetailResult	LoadDetail	
Error[]	Errors	

class MultipleLoadDetailReturn

MultipleLoadDetailResults	DetailResults	(Sorted by ENTERED descending)
Error[]	Errors	

class LoadDetailResult

int	ID	
string	MCNumber	
string	TMCNumber	
string	DOTNumber	
string	OriginCity	
string	OriginState	
string	OriginZip	
string	DestinationCity	
string	DestinationState	
string	DestinationZip	
string	Weight	
string	Length	
EquipmentTypes	EquipmentTypes	
TrailerOptionType[]	EquipmentOptions	
string	SpecInfo	
LoadType	LoadType	
string	Quantity	
string	Width	
string	PointOfContact	
string	PointOfContactPhone	
DateTime	Entered	
string	TruckCompanyName	
string	TruckCompanyEmail	
string	TruckCompanyPhone	
string	TruckCompanyFax	
string	TruckCompanyCity	
string	TruckCompanyState	
string	Distance	
string	Stops	
string	PaymentAmount	
string	PickupDate	
string	PickupTime	
string	DeliveryDate	
string	DeliveryTime	
string	Credit	– VA
string	Mileage	– VA
string	FuelCost	– VA

int	Bond	– see Appendix 5 – Bond Information for details
bool	HasBonding	– see Appendix 5 – Bond Information for details
int	BondTypeID	– see Appendix 5 – Bond Information for details
int	DeletedId	
bool	IsDeleted	
string	HandleName	

class MultipleLoadDetailResult

int	ID	
string	MCNumber	
string	TMCNumber	
string	DOTNumber	
string	OriginCity	
string	OriginState	
string	OriginZip	
string	DestinationCity	
string	DestinationState	
string	DestinationZip	
string	Weight	
string	Length	
EquipmentTypes	EquipmentTypes	
TrailerOptionType[]	EquipmentOptions	
string	SpecInfo	
LoadType	LoadType	
string	Quantity	
string	Width	
string	PointOfContact	
string	PointOfContactPhone	
DateTime	Entered	
string	TruckCompanyName	
string	TruckCompanyEmail	
string	TruckCompanyPhone	
string	TruckCompanyFax	
string	TruckCompanyCity	
string	TruckCompanyState	
string	Distance	
string	Stops	
string	PaymentAmount	
string	PickupDate	
string	PickupTime	
string	DeliveryDate	
string	DeliveryTime	
string	Credit – VA	

string	Mileage – VA	
string	FuelCost – VA	
int	Bond	– see Appendix 5 – Bond Information for details
bool	HasBonding	– see Appendix 5 – Bond Information for details
int	BondTypeID	– see Appendix 5 – Bond Information for details
int	DeletedId	
bool	IsDeleted	
string	HandleName	
string	OriginCountry	
string	DestinationCountry	
string	Equipment	
string	ExperienceFactor	
decimal	PricePerGallon	
bool	IsFriend	
string	Age	
int	OriginDistance	
int	DestinationDistance	

class Error

string	ErrorMessage	
--------	--------------	--

class EquipmentTypes

bool	IsCombo	
int	MapTold	
bool	IsTruckPost	
int	Id	
string	Code	
string	Description	
string	Category	
int	CategoryId	
bool	IsCategorizable	
bool	FullLoad	
TrailerOptionType	RequiredOption	
bool	WebserviceOnly	

enum LoadType

0. Nothing
1. All
2. Full
3. Partial

enum TrailerOptionType

1. Tarps

2. Hazardous
3. PalletExchange
4. Team
5. Expedited

enum SortColumns

1. Equipment
2. LoadType
3. PickupDate
4. OriginCity
5. OriginState
6. OriginDistance
7. DestinationCity
8. DestinationState
9. Payment
10. PaymentAmount
11. Length
12. Weight
13. Days2Pay
14. Credit
15. ExperienceFactor
16. CompanyName
17. TruckCompanyName
18. FuelCost
19. Miles
20. Mileage
21. Age

Appendix 2 – Supported Equipment Types

2F	- Two 24 or 28 foot flats
ANIM	- Animal Carrier
ANY	- Any Equipment
AUTO	- Auto Carrier
B-TR	- B-Train/Supertrain (Canada only)
BDMP	- Belly Dump
BEAM	- Beam
BELT	- Conveyor Belt
BOAT	- Boat Hauling Trailer

CH	- Convertible Hopper
CONG	- Conestoga
CONT	- Container Trailer
CV	- Curtain Van
DA	- Drive Away
DD	- Double Drop
DDE	- Double Drop Extendable
DUMP	- Dump Trucks
ENDP	- End Dump
F	- Flatbed
FA	- Flatbed Air-Ride
FEXT	- Stretch Trailers or Extendable Flatbed
FINT	- Flatbed Intermodal
FO	- Flatbed over-dimension loads
FRV	- Flatbed, Van, or Reefer
FSD	- Flat or Step Deck
FSDV	- Flatbed, Step Deck, or Van
FV	- Van or Flatbed
FVR	- Flatbed, Van, or Reefer
FVV	- Flatbed, or Vented Van
FVVR	- Flatbed, Vented Van or Reefer
FWS	- Flatbed With Sides
HOPP	- Hopper Bottom
HS	- Hot Shot
HTU	- Haul and Tow Unit
LAF	- Landoll Flatbed
LB	- Lowboy
LBO	- Lowboy over-dimension load
LDOT	- Load-Out are empty trailers you load and haul
LIVE	- Live Bottom Trailer
MAXI	- Maxi or Double Flat Trailers
MBHM	- Mobile Home
PNEU	- Pneumatic
PO	- Power Only (tow-away)
R	- Refrigerated Carrier(Reefer)
RFV	- Flatbed, Van, or Reefer
RGN	- Removable Goose Neck & Multi-Axle Heavy Haulers
RGNE	- RGN Extendable
RINT	- Refrigerated Intermodal
ROLL	- Roll Top Conestoga
RPD	- Refrigerated Carrier W/Plant Decking
RV	- Van or Reefer

RVF	- Flatbed, Van or Reefer
SD	- Step Deck
SDL	- Step Deck w/Loading Ramps
SDO	- Step Deck over-dimension load
SDRG	- Step Deck or Removable Gooseneck
SPEC	- Unspecified Specialized Trailers
SV	- Straight Van
TANK	- Tanker (Please specify Food grade, liquid, bulk, etc. in your posting)
V	- Van
V-OT	- Open Top Van
VA	- Van Air-Ride
VB	- Blanket Wrap Van
VCAR	- Cargo Vans (1 Ton capacity)
VF	- Flatbed or Van
VFR	- Flatbed, Van, or Reefer
VINT	- Van Intermodal
VIV	- Vented Insulated Van
VIVR	- Vented Insulated Van or Refrigerated
VLG	- Van w/Liftgate
VM	- Moving Van
VR	- Van or Reefer
VRDD	- Van, Reefer Double Drop
VRF	- Flatbed, Van, or Reefer
VV	- Vented Van
VVR	- Vented Van or Refrigerated
WALK	- Walking Floor

Appendix 3 – Supported Countries

USA	- United States
CAN	- Canada
MEX	- Mexico

Appendix 4 - Supported States and Provinces

United States

AL	- ALABAMA
AK	- ALASKA
AZ	- ARIZONA
AR	- ARKANSAS
CA	- CALIFORNIA

CO	- COLORADO
CT	- CONNECTICUT
DE	- DELAWARE
DC	- DISTRICT OF COLUMBIA
FL	- FLORIDA
GA	- GEORGIA
HI	- HAWAII
ID	- IDAHO
IL	- ILLINOIS
IN	- INDIANA
IA	- IOWA
KS	- KANSAS
KY	- KENTUCKY
LA	- LOUISIANA
MA	- MAINE
ME	- MAINE
MD	- MARYLAND
MI	- MICHIGAN
MN	- MINNESOTA
MS	- MISSISSIPPI
MO	- MISSOURI
MT	- MONTANA
NE	- NEBRASKA
NV	- NEVADA
NH	- NEW HAMPSHIRE
NJ	- NEW JERSEY
NM	- NEW MEXICO
NY	- NEW YORK
NC	- NORTH CAROLINA
ND	- NORTH DAKOTA
OH	- OHIO
OK	- OKLAHOMA
OR	- OREGON
PA	- PENNSYLVANIA
RI	- RHODE ISLAND
SC	- SOUTH CAROLINA
SD	- SOUTH DAKOTA
TN	- TENNESSEE
TX	- TEXAS
UT	- UTAH
VT	- VERMONT
VA	- VIRGINIA

WA	- WASHINGTON
WV	- WEST VIRGINIA
WI	- WISCONSIN
WY	- WYOMING

Canada

AB	- ALBERTA
BC	- BRITISH COLUMBIA
MB	- MANITOBA
NB	- NEW BRUNSWICK
NL	- NEWFOUNDLAND AND LABRADOR
NS	- NOVA SCOTIA
NT	- NORTHWEST TERRITORIES
ON	- ONTARIO
PE	- PRINCE EDWARD ISLAND
QC	- QUEBEC
SK	- SASKATCHEWAN
YT	- YUKON

Mexico

AG	- AGUASCALIENTES
BC	- BAJA CALIFORNIA NORTE
BS	- BAJA CALIFORNIA SUR
CH	- CHIHUAHUA
CL	- COLIMA
CM	- CAMPECHE
CO	- COAHUILA
CS	- CHIAPAS
DF	- DISTRITO FEDERAL
DG	- DURANGO
GR	- GUERRERO
GT	- GUANAJUATO
HG	- HIDALGO
JA	- JALISCO
MI	- MICHOACAN
MO	- MORELOS
MX	- MEXICO
NA	- NAYARIT
NL	- NUEVO LEON
OA	- OAXACA
PU	- PUEBLA
QR	- QUINTANA ROO

QT	- QUERETARO
SI	- SINALOA
SL	- SAN LUIS POTOSI
SO	- SONORA
TB	- TABASCO
TL	- TLAXCALA
TM	- TAMAULIPAS
VE	- VERACRUZ
YU	- YUCATAN
ZT	- ZACATECAS

Appendix 5 – Bond Information

BondTypeID used with Bond will provide the correct Bond amount for the broker in the record returned.

The values of Bond can be 10, 25, 50, 100, and 250. All values are in \$1000 increments, meaning 10 represents \$10,000, 25 represents \$25,000,

There are three bond types designated by color:

Red - Brokerage is a participating member of the Diamond Broker program at level I (highest level).

Blue - This Brokerage is a participating member of the TIA's Performance Certified program.

Gray - This Brokerage is a participating member of the Diamond Broker program.

The following define what each BondTypeID represents:

Bond at 10K Red (1 diamond) - 105

Bond at 25K Red (2 diamonds) - 104

Bond at 50K Red (3 diamonds) - 103

Bond at 100K Red (4 diamonds) - 102

Bond at 250K Red (5 diamonds) – 101

Bond at 10K Red (1 diamond) – 205 (Trust not held by ITSFS, but still verified)

Bond at 10K Blue (1 diamond) - 505

Bond at 25K Blue (2 diamonds) - 504

Bond at 50K Blue (3 diamonds) - 503

Bond at 100K Blue (4 diamonds) - 502

Bond at 250K Blue (5 diamonds) - 501

Bond at 10K Gray (1 diamond) - 1005

Bond at 25K Gray (2 diamonds) - 1004

Bond at 50K Gray (3 diamonds) - 1003

Bond at 100K Gray (4 diamonds) - 1002

Bond at 250K Gray (5 diamonds) - 1001

No Bond - 9999

The BondTypeID numbering is for proper sorting.

Appendix 6 – XML Examples

GetLoadSearchResults XML

POST http://webservices.truckstop.com/V13/Searching/LoadSearch.svc HTTP/1.1

Accept-Encoding: gzip,deflate

Content-Type: text/xml; charset=UTF-8

SOAPAction: "http://webservices.truckstop.com/v12/ILoadSearch/GetLoadSearchResults"

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:v12="http://webservices.truckstop.com/v12"
xmlns:web="http://schemas.datacontract.org/2004/07/WebServices"
xmlns:web1="http://schemas.datacontract.org/2004/07/WebServices.Searching"
xmlns:truc="http://schemas.datacontract.org/2004/07/Truckstop2.Objects"
xmlns:arr="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <soapenv:Header/>
  <soapenv:Body>
    <v12:GetLoadSearchResults>
      <v12:searchRequest>
        <web:IntegrationId>###</web:IntegrationId>
        <web:Password>password</web:Password>
        <web:UserName>username</web:UserName>
        <web1:Criteria>
          <web1:DestinationCity>Dallas</web1:DestinationCity>
          <web1:DestinationCountry>USA</web1:DestinationCountry>
          <web1:DestinationRange>100</web1:DestinationRange>
          <web1:DestinationState>TX</web1:DestinationState>
          <web1:EquipmentOptions>
            <truc:TrailerOptionType></truc:TrailerOptionType>
          </web1:EquipmentOptions>
          <web1:EquipmentType>F</web1:EquipmentType>
          <web1:HoursOld>0</web1:HoursOld>
          <web1:LoadType>All</web1:LoadType>
          <web1:OriginCity>Boise</web1:OriginCity>
          <web1:OriginCountry>USA</web1:OriginCountry>
          <web1:OriginLatitude>0</web1:OriginLatitude>
          <web1:OriginLongitude>0</web1:OriginLongitude>
          <web1:OriginRange>100</web1:OriginRange>
          <web1:OriginState>ID</web1:OriginState>
          <web1:PageNumber>1</web1:PageNumber>
          <web1:PageSize>50</web1:PageSize>
          <web1:PickupDates>
            <arr:dateTime>0001-01-01</arr:dateTime>
          </web1:PickupDates>
          <web1:SortBy>Age</web1:SortBy>
          <web1:SortDescending>false</web1:SortDescending>
        </web1:Criteria>
      </v12:searchRequest>
    </v12:GetLoadSearchResults>
  </soapenv:Body>
</soapenv:Envelope>
```

HTTP/1.1 200 OK
Content-Length: 1062
Content-Type: text/xml; charset=utf-8
Server: Microsoft-IIS/7.0
X-Powered-By: ASP.NET
Date: Mon, 09 Apr 2012 19:36:52 GMT
Vary: Accept-Encoding
Content-Encoding: gzip
Connection: Keep-Alive

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <ActivityId CorrelationId="ec90c39f-f527-4353-8a34-e0abb5c3378"
xmlns="http://schemas.microsoft.com/2004/09/ServiceModel/Diagnostics">96cafad8-a5b8-4ce6-80a1-40ad2d24922b</ActivityId>
  </s:Header>
  <s:Body>
    <GetLoadSearchResultsResponse xmlns="http://webservices.truckstop.com/v12">
      <GetLoadSearchResultsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebServices.Objects"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Errors xmlns="http://schemas.datacontract.org/2004/07/WebServices"/>
        <a:SearchResults>
          <a:LoadSearchItem>
            <a:Age>0</a:Age>
            <a:Bond>0</a:Bond>
            <a:BondEnabled>>false</a:BondEnabled>
            <a:BondTypeID>9999</a:BondTypeID>
            <a:CompanyName>Company</a:CompanyName>
            <a:Days2Pay>info</a:Days2Pay>
            <a:DestinationCity>WACO</a:DestinationCity>
            <a:DestinationCountry>USA</a:DestinationCountry>
            <a:DestinationDistance>88</a:DestinationDistance>
            <a:DestinationState>TX</a:DestinationState>
            <a:Equipment>F</a:Equipment>
            <a:EquipmentOptions xmlns:b="http://schemas.datacontract.org/2004/07/Truckstop2.Objects"/>
            <a:ExperienceFactor>N</a:ExperienceFactor>
            <a:FuelCost>4.0786</a:FuelCost>
            <a:ID>111111111</a:ID>
            <a:IsFriend>>false</a:IsFriend>
            <a:Length>0</a:Length>
            <a:LoadType>Full</a:LoadType>
            <a:Miles>1695</a:Miles>
            <a:OriginCity>BOISE</a:OriginCity>
            <a:OriginCountry>USA</a:OriginCountry>
            <a:OriginDistance>0</a:OriginDistance>
            <a:OriginState>ID</a:OriginState>
            <a:Payment>0</a:Payment>
            <a:PickUpDate>5/13/2013</a:PickUpDate>
            <a:PointOfContactPhone>123-456-7891</a:PointOfContactPhone>
            <a:PricePerGall>0</a:PricePerGall>
            <a:Weight>48,000</a:Weight>
          </a:LoadSearchItem>
        </a:SearchResults>
      </GetLoadSearchResultsResult>
    </GetLoadSearchResultsResponse>
  </s:Body>
</s:Envelope>
```

```
</s:Body>
</s:Envelope>
```

GetLoadSearchDetailResult XML

POST http://webservices.truckstop.com/V13/Searching/LoadSearch.svc HTTP/1.1
Accept-Encoding: gzip,deflate
Content-Type: text/xml; charset=UTF-8
SOAPAction: "http://webservices.truckstop.com/v12/ILoadSearch/GetLoadSearchDetailResult"

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:v12="http://webservices.truckstop.com/v12"
xmlns:web="http://schemas.datacontract.org/2004/07/WebServices"
xmlns:web1="http://schemas.datacontract.org/2004/07/WebServices.Searching">
  <soapenv:Header/>
  <soapenv:Body>
    <v12:GetLoadSearchDetailResult>
      <v12:detailRequest>
        <web:IntegrationId>###</web:IntegrationId>
        <web:Password>password</web:Password>
        <web:UserName>username</web:UserName>
        <web1:LoadId>11111111</web1:LoadId>
      </v12:detailRequest>
    </v12:GetLoadSearchDetailResult>
  </soapenv:Body>
</soapenv:Envelope>
```

HTTP/1.1 200 OK
Content-Length: 1165
Content-Type: text/xml; charset=utf-8
Server: Microsoft-IIS/7.0
X-Powered-By: ASP.NET
Date: Mon, 09 Apr 2012 19:41:58 GMT
Vary: Accept-Encoding
Content-Encoding: gzip
Connection: Keep-Alive

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <ActivityId CorrelationId="e39129c9-8b20-413a-8903-4bf202f2c00f"
xmlns="http://schemas.microsoft.com/2004/09/ServiceModel/Diagnostics">9c80eef5-b9aa-444c-ba93-6e89ba48e362</ActivityId>
  </s:Header>
  <s:Body>
    <GetLoadSearchDetailResultResponse xmlns="http://webservices.truckstop.com/v12">
      <GetLoadSearchDetailResultResult xmlns:a="http://schemas.datacontract.org/2004/07/WebServices.Objects"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Errors xmlns="http://schemas.datacontract.org/2004/07/WebServices"/>
        <a:LoadDetail>
          <a:Bond>0</a:Bond>
          <a:BondTypeID>9999</a:BondTypeID>
          <a:Credit>info</a:Credit>
          <a:DOTNumber>#####</a:DOTNumber>
          <a:DeletedId>0</a:DeletedId>
          <a:DeliveryDate></a:DeliveryDate>
          <a:DeliveryTime></a:DeliveryTime>
          <a:DestinationCity>WACO</a:DestinationCity>
          <a:DestinationState>TX</a:DestinationState>
```

```
<a:DestinationZip>76701</a:DestinationZip>
<a:Distance></a:Distance>
<a:Entered>2012-04-09T13:32:27</a:Entered>
<a:EquipmentOptions xmlns:b="http://schemas.datacontract.org/2004/07/Truckstop2.Objects"/>
<a:EquipmentTypes xmlns:b="http://schemas.datacontract.org/2004/07/Truckstop2.Objects">
  <b:Category>F</b:Category>
  <b:CategoryId>1</b:CategoryId>
  <b:Code>F</b:Code>
  <b>Description>Flatbed</b>Description>
  <b:Id>12</b:Id>
  <b:IsCategorizable>true</b:IsCategorizable>
  <b:IsCombo>false</b:IsCombo>
  <b:IsTruckPost>true</b:IsTruckPost>
  <b:MapTold>12</b:MapTold>
  <b:RequiredOption i:nil="true"/>
  <b:WebserviceOnly>false</b:WebserviceOnly>
</a:EquipmentTypes>
<a:FuelCost>$1,257.00</a:FuelCost>
<a:HasBonding>false</a:HasBonding>
<a:ID>11111111</a:ID>
<a:IsDeleted>false</a:IsDeleted>
<a:Length>0</a:Length>
<a:LoadType>Full</a:LoadType>
<a:MCNumber>#####</a:MCNumber>
<a:Mileage>1695</a:Mileage>
<a:OriginCity>BOISE</a:OriginCity>
<a:OriginState>ID</a:OriginState>
<a:OriginZip>83701</a:OriginZip>
<a:PaymentAmount>0</a:PaymentAmount>
<a:PickupDate>5/13/13</a:PickupDate>
<a:PickupTime></a:PickupTime>
<a:PointOfContact>DISPATCH</a:PointOfContact>
<a:PointOfContactPhone>123-456-7891</a:PointOfContactPhone>
<a:Quantity>1</a:Quantity>
<a:SpecInfo></a:SpecInfo>
<a:Stops>0</a:Stops>
<a:TMCNumber/>
<a:TruckCompanyCity>New Plymouth</a:TruckCompanyCity>
<a:TruckCompanyEmail>nobody@truckstop.com</a:TruckCompanyEmail>
<a:TruckCompanyFax>123-456-7891</a:TruckCompanyFax>
<a:TruckCompanyName>Company</a:TruckCompanyName>
<a:TruckCompanyPhone>123-456-7891</a:TruckCompanyPhone>
<a:TruckCompanyState>ID</a:TruckCompanyState>
<a:Weight>48,000</a:Weight>
<a:Width>0</a:Width>
</a:LoadDetail>
</GetLoadSearchDetailResultResult>
</GetLoadSearchDetailResultResponse>
</s:Body>
</s:Envelope>
```

GetMultipleLoadDetailResults XML

POST http://webservices.truckstop.com/V13/Searching/LoadSearch.svc HTTP/1.1

Accept-Encoding: gzip, deflate

Content-Type: text/xml; charset=UTF-8

SOAPAction: "http://webservices.truckstop.com/v12/ILoadSearch/GetMultipleLoadDetailResults"

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:v12="http://webservices.truckstop.com/v12"
xmlns:web="http://schemas.datacontract.org/2004/07/WebServices"
xmlns:web1="http://schemas.datacontract.org/2004/07/WebServices.Searching"
xmlns:truc="http://schemas.datacontract.org/2004/07/Truckstop2.Objects"
xmlns:arr="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <soapenv:Header/>
  <soapenv:Body>
    <v12:GetMultipleLoadDetailResults>
      <v12:searchRequest>
        <web:IntegrationId>###</web:IntegrationId>
        <web:Password>password</web:Password>
        <web:UserName>username</web:UserName>
        <web1:Criteria>
          <web1:DestinationCity></web1:DestinationCity>
          <web1:DestinationCountry></web1:DestinationCountry>
          <web1:DestinationRange>0</web1:DestinationRange>
          <web1:DestinationState></web1:DestinationState>
          <web1:EquipmentOptions>
            <truc:TrailerOptionType>Tarps</truc:TrailerOptionType>
          </web1:EquipmentOptions>
          <web1:EquipmentType>VFR</web1:EquipmentType>
          <web1:HoursOld>0</web1:HoursOld>
          <web1:LoadType>All</web1:LoadType>
          <web1:OriginCity>Denver</web1:OriginCity>
          <web1:OriginCountry>USA</web1:OriginCountry>
          <web1:OriginLatitude>0</web1:OriginLatitude>
          <web1:OriginLongitude>0</web1:OriginLongitude>
          <web1:OriginRange>100</web1:OriginRange>
          <web1:OriginState>CO</web1:OriginState>
          <web1:PageNumber>1</web1:PageNumber>
          <web1:PageSize>1</web1:PageSize>
          <web1:PickupDates>
            <arr:dateTime>2014-02-05</arr:dateTime>
          </web1:PickupDates>
          <web1:SortBy>Age</web1:SortBy>
          <web1:SortDescending>false</web1:SortDescending>
        </web1:Criteria>
      </v12:searchRequest>
    </v12:GetMultipleLoadDetailResults>
  </soapenv:Body>
</soapenv:Envelope>
```

HTTP/1.1 200 OK

Content-Length: 1390

Content-Type: text/xml; charset=utf-8

Server: Microsoft-IIS/7.0

X-Powered-By: ASP.NET

Date: Mon, 09 Apr 2012 19:46:44 GMT

Vary: Accept-Encoding

Content-Encoding: gzip
Connection: Keep-Alive

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <ActivityId CorrelationId="8ab97995-9342-4a61-8d23-c08b26febcc0"
xmlns="http://schemas.microsoft.com/2004/09/ServiceModel/Diagnostics">c59ff261-8b44-4f81-9bd7-4658232e8be7</ActivityId>
  </s:Header>
  <s:Body>
    <GetMultipleLoadDetailResultsResponse xmlns="http://webservices.truckstop.com/v12">
      <GetMultipleLoadDetailResultsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebServices.Objects"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Errors xmlns="http://schemas.datacontract.org/2004/07/WebServices"/>
        <a:DetailResults>
          <a:MultipleLoadDetailResult>
            <a:Age>9+</a:Age>
            <a:Bond>75</a:Bond>
            <a:BondTypeID>103</a:BondTypeID>
            <a:Credit>34</a:Credit>
            <a:DOTNumber>#####</a:DOTNumber>
            <a:DeletedId>0</a:DeletedId>
            <a:DeliveryDate/>
            <a:DeliveryTime/>
            <a:DestinationCity>SPRINGDALE</a:DestinationCity>
            <a:DestinationCountry>USA</a:DestinationCountry>
            <a:DestinationDistance>0</a:DestinationDistance>
            <a:DestinationState>AR</a:DestinationState>
            <a:DestinationZip>72762</a:DestinationZip>
            <a:Distance>510</a:Distance>
            <a:Entered>2014-05-22T12:24:16.04</a:Entered>
            <a:Equipment>F</a:Equipment>
            <a:EquipmentOptions xmlns:b="http://schemas.datacontract.org/2004/07/Truckstop2.Objects"/>
            <a:EquipmentTypes xmlns:b="http://schemas.datacontract.org/2004/07/Truckstop2.Objects">
              <b:Category>F</b:Category>
              <b:CategoryId>1</b:CategoryId>
              <b:Code>F</b:Code>
              <b:Description>Flatbed</b:Description>
              <b:FullLoad i:nil="true"/>
              <b:Id>12</b:Id>
              <b:IsCategorizable>true</b:IsCategorizable>
              <b:IsCombo>false</b:IsCombo>
              <b:IsTruckPost>true</b:IsTruckPost>
              <b:MapTold>12</b:MapTold>
              <b:RequiredOption i:nil="true"/>
              <b:WebserviceOnly>false</b:WebserviceOnly>
            </a:EquipmentTypes>
            <a:ExperienceFactor>A</a:ExperienceFactor>
            <a:FuelCost/>
            <a:HandleName>Dispatch</a:HandleName>
            <a:HasBonding>true</a:HasBonding>
            <a:ID>823025694</a:ID>
            <a:IsDeleted>false</a:IsDeleted>
            <a:IsFriend>false</a:IsFriend>
            <a:Length>48</a:Length>
            <a:LoadType>Full</a:LoadType>
            <a:MCNumber>#####</a:MCNumber>
            <a:Mileage>0</a:Mileage>
          </a:MultipleLoadDetailResult>
        </a:DetailResults>
      </GetMultipleLoadDetailResultsResult>
    </GetMultipleLoadDetailResultsResponse>
  </s:Body>
</s:Envelope>
```

```
<a:OriginCity>BRYAN</a:OriginCity>
<a:OriginCountry>USA</a:OriginCountry>
<a:OriginDistance>143</a:OriginDistance>
<a:OriginState>TX</a:OriginState>
<a:OriginZip>77801</a:OriginZip>
<a:PaymentAmount>900</a:PaymentAmount>
<a:PickupDate>DAILY</a:PickupDate>
<a:PickupTime/>
<a:PointOfContact>Dispatch<a:PointOfContact>
<a:PointOfContactPhone>123-456-7891</a:PointOfContactPhone>
<a:PricePerGallon>0.0000</a:PricePerGallon>
<a:Quantity>1</a:Quantity>
<a:SpecInfo>LUMBER</a:SpecInfo>
<a:Stops>0</a:Stops>
<a:TMCNumber>#####</a:TMCNumber>
<a:TruckCompanyCity>GREENVILLE</a:TruckCompanyCity>
<a:TruckCompanyEmail>nobody@truckstop.com</a:TruckCompanyEmail>
<a:TruckCompanyFax>123-456-7891<a:TruckCompanyFax>
<a:TruckCompanyName>TENNESSE</a:TruckCompanyName>
<a:TruckCompanyPhone>270-338-7969</a:TruckCompanyPhone>
<a:TruckCompanyState>KY</a:TruckCompanyState>
<a:Weight>48000</a:Weight>
<a:Width>6.00</a:Width>
</a:MultipleLoadDetailResult>
</a:DetailResults>
</GetMultipleLoadDetailResultsResult>
</GetMultipleLoadDetailResultsResponse>
</s:Body>
</s:Envelope>
```